

BOSS

As electronic systems become more complex, performance evaluations and tradeoff studies based solely on analytical techniques become unfeasible. "Breadboarding" all the hardware components of a large system is usually out of the question for reasons of schedule and cost. As a result, computer simulation is assuming greater importance as a flexible and expedient approach to modeling system and subsystem behavior.

Simulation has played a key role in the growth of complex, multiple access space communications such as those used by the space shuttle and the TRW-built Tracking and Data Relay Satellites (TDRS). Today, under a National Aeronautics and Space Administration (NASA) contract awarded in August 1986, the System Engineering Laboratory (SEL) of ESG's Space Communications Division (SCD) is developing a powerful new simulator for use in designing and modeling the communication system of NASA's planned Space Station (Figure 1').

Taking advantage of 1980s advances in software development and interactive graphics, the Space Station Communications System Simulator (SCSS) boasts such features as pop-up menus, "windows" (which allow simultaneous manipulation of multiple data on the same screen), "mouse" pointing, and on-line help information. Not only that, but SCSS automatically generates Fortran computer code for simulation models, which the user enters in block diagram form. By project completion in 1988, SCSS will furnish NASA with "unparalleled simulation power, flexibility, user friendliness, and cost

N88-27760

(NASA-CR-182947) BLOCK ORIENTED SIMULATION

SYSTEM (BOSS) (TRW Defense and Space Systems Group) 45 p CSCI 09B

Unclas
G3/61 0146672

effectiveness," said SCSS Project Manager Tom Manning.

TRW's win of the SCSS contract stems from an ongoing independent research and development (IR&D) project begun in 1983 in conjunction with the University of Kansas for the purpose of developing a general purpose Block [Diagram] Oriented Simulation System, or BOSS, as an upgrade to existing simulators (see box, page XX). The SCSS to be delivered to NASA's Johnson Space Center consists of BOSS, which serves as the operating system and determines the simulation architecture; an interactive graphics workstation hosting SCSS, that will interface with the Space Center's computers; a library of hardware models for use in Space Station simulations; and source codes and documentation.

"Three years ago, when we realized that TRW was not the leader in systems simulation, we made an IR&D investment to strengthen our simulation technology and capabilities. In 1986, in the SCSS proposal we went up against the top names in simulation -- Hughes, RCA, Harris, the Lincoln Lab -- and came out with a contract because we were technically superior. That shows what an IR&D investment can do," said Raul Rey, Manager of SEL's Communication Systems Engineering Department.

The incentive to develop an updated simulator arose when SEL engineers began to chafe at the restrictions imposed by widely used communications simulation packages such as LINK (developed by TRW during TDRS design); ICSSM, developed by the Hazeltine Corporation for the Air Force; ICS, developed by Rensselaer Polytechnic Institute for the Air Force and marketed in modified form by Par Technology Corporation as CMS; and Hughes

Aircraft Company's SYSTID. Developed during the 1970s, these simulators lack features available in today's hardware and software. Many, for example, run on a mainframe computer with user input through a simple terminal or a personal computer, an approach that precludes the use of high-speed graphics user interfaces. Similarly, LINK continues as an excellent tool for TRW's ongoing TDRS program; but in attempting to adapt it for other purposes, "we had patched it so often it couldn't be patched any more," Manning said.

By 1983, attempts to adapt LINK for more general capabilities came to an end. A 1984 IR&D project, headed by Dr. Eric Wiswell, now a senior staff analyst in the Defense Communications Division's (DCD) Defense Systems Engineering Laboratory, was funded to begin development of an entirely new general-purpose simulator based on 1980s technologies. "We started by asking 'What properties do we want in a simulator?'" said Wiswell. "After consultation with system engineers, hardware designers, and software houses, we came up with five qualities: understandability, utility, flexibility, expandability, and portability" (Figure 2) These five qualities became system definition drivers shaping the new simulator. Throughout 1984 and 1985, the IR&D team collaborated closely with Dr. Sam Shanmugan and software experts at the University of Kansas on conceptual design of the new system, assessing various architectures and evaluating hardware.

"At the final design review of the completed architecture, held in Kansas in late 1985, we felt we had met all five of our goals," Wiswell said. By February 1986, when NASA issued a Request for Proposal for the SCSS, BOSS

was rapidly emerging as an integrated simulation workstation able to take advantage of recent advances in simulation techniques (such as modified Monte Carlo procedures that reduce time required for simulation), software engineering, workstation technology, CAD/CAM techniques, and artificial intelligence/expert systems. "During the SCSS proposal writing stage, we were actually able to model Space Station components on BOSS and include them as examples," Manning said.

As the operating system for simulation, BOSS oversees all the software functions necessary to build models, run simulations, and view simulation results. BOSS applies a functional approach to simulation (i.e., user-specified functions, or blocks, operate on current inputs to produce current outputs).

BOSS assumes that modules, subsystems, and systems to be simulated can be represented in a hierarchical block diagram form. Blocks used to configure a communications link are drawn from a library of subsystem models, such as encoders, modulators, or noise sources. These models are software routines which simulate the signal processing operations that take place in each functional link. Once the system is configured, the simulation exerciser performs a waveform level simulation of the system in time domain and stores waveforms from selected points in the system. Following the execution phase, the post processor is used to evaluate and view the results of the simulation.

Unlike earlier simulators, BOSS' interactive environment offers greatly enhanced capability in model construction, system configuration,

simulation, and results analysis. These functions are carried out by a modular software package consisting of six components: a Display Manager, Block Diagram Editor, Database Manager, Code Generator, Simulation Manager, and Post Processor (Figure 3).

The Display Manager maintains a large number of windows on the display so that the user can obtain and examine information from several model levels at the same time. It also maintains the menu system used to select operations and to select and specify models. Users employ a mouse to select editing operations, indicate positions, and select items from a menu; only textual data requires use of the terminal keyboard.

The Database Manager handles storage and retrieval of model definitions and simulation results from files on mass storage devices while supporting access to several model databases concurrently.

The Block Diagram Editor uses the Display Manager and obtains model definitions from the Database Manager. The Editor encourages a hierarchical, bottom-up design approach, since to define a new module the engineer selects from a menu of existing modules. At the lowest level, the BOSS library contains a set of primitive modules, such as adders, multipliers, and comparators. The engineer then connects selected modules with signals and provides required parameters and documentation. The new module thus created is then stored into a database for recall as a single functional block at a higher level in system design. This approach allows rapid synthesizing of increasingly complex modules or systems through use of previously defined modules.

SCSS Schedule

1986	1987	1988
S O N D	J F M A M J J A S O N D	J F M A M J J A

Δ contract start

Architecture Definition

Δ VAXstation-II delivery to JSC

Δ Design Review

Model Development

Δ training class

Δ final delivery

(software and documentation)

Figure 2. BOSS' Five General Characteristics

The Terminology of Desirability

Understandability - The simulator must be useful to those with minimal knowledge of its internal structure. Documentation must include a clearly written user's manual for less sophisticated users, as well as technical documentation sufficient for the most detailed studies.

Utility - The simulator must be easy to use. The operator/machine interface must be able to translate user input into an executable simulation program, allowing users detachment from the actual simulation.

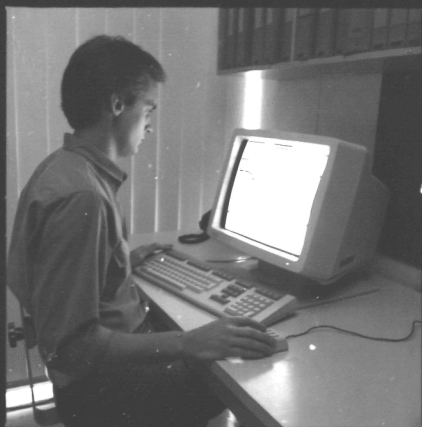
Flexibility - The simulator must have enough topological flexibility to represent modern and future communications systems. The capability to creatively reconfigure communications systems under study greatly aids design tradeoff analysis.

Expandability - The simulator must be easily expandable. This ensures that it will not become obsolete for modeling future communications systems.

Portability - The simulator must be easily transportable from one engineering development facility to others.

(Copy to be overprinted on photo of Jerry Lacey
~~at~~ working at BOSS terminal. TBS)

~~Handwritten mark~~



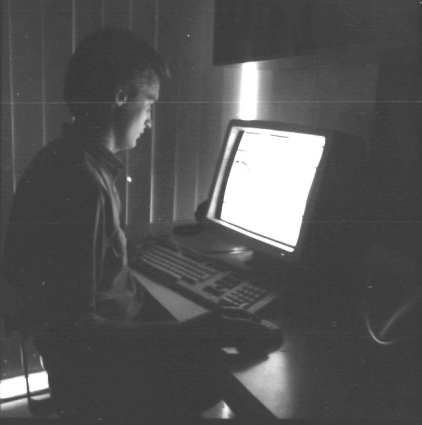
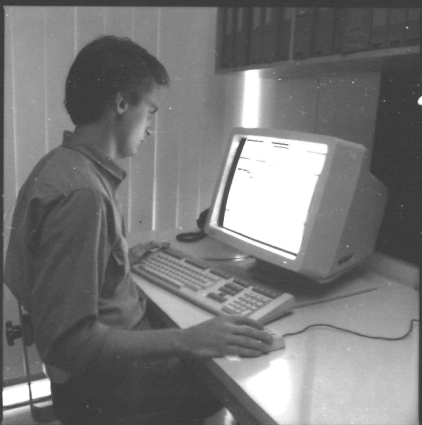
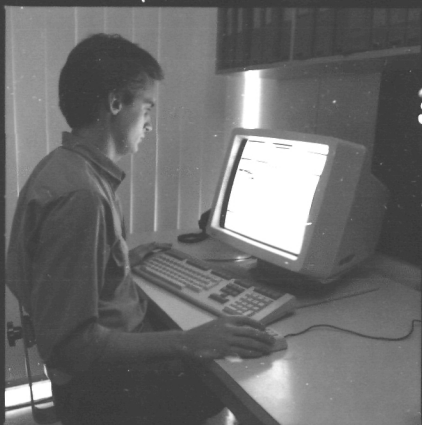
216916 - 86



216916 - 86



216916 - 86



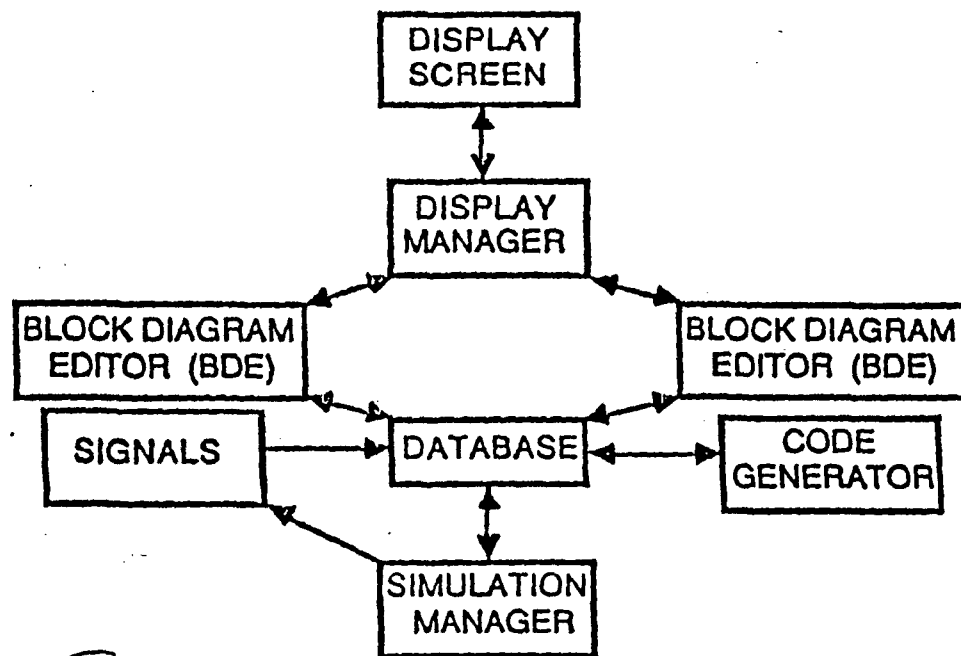


Figure 3.

Figure 2-3. BOSS Modular Software Structure Uses Windows and a Mouse to Encourage User Interaction

The Code Generator is responsible for converting model block diagrams created with the Block Diagram Editor into a FORTRAN subroutine. The Generator is capable of recognizing each submodel and allocates necessary storage so that all models are reentrant for multiple use in higher level specifications.

'BOSS' code-generating block diagram method is a "natural way" to approach simulation, Wiswell said. "The first thing a system engineer does in a simulation is to draw a block diagram. Instead of drawing the diagram on paper, then sitting down at a computer terminal and typing code-like statements, the engineer can now sit down at the BOSS terminal, use the mouse to create his block diagram, and let the computer generate the code statements. The computer writes codes far more quickly and accurately than most engineers can."

The Simulation Manager ensures that all input signals and parameters of a model have been defined by prompting the user for undefined simulation inputs. It then runs the simulation on the local workstation or a remote computer.

Finally, the Post Processor has access to the block diagram of the simulated model, allowing the user to select a signal from the diagram by simply pointing with the mouse to the signal node. The Processor reads from the database the signals saved during the simulation and displays them according to user-selected display modes that include frequency spectrum, time domain waveform, eye diagram, power level data throughout a channel, signal-to-noise ratio, X-Y plots of any two vectors, bit error rate (BER)

sensitivity analysis, carrier acquisition data (e.g., phase error vs. time), and bit timing acquisition data, such as timing error vs. time (Figure 4).

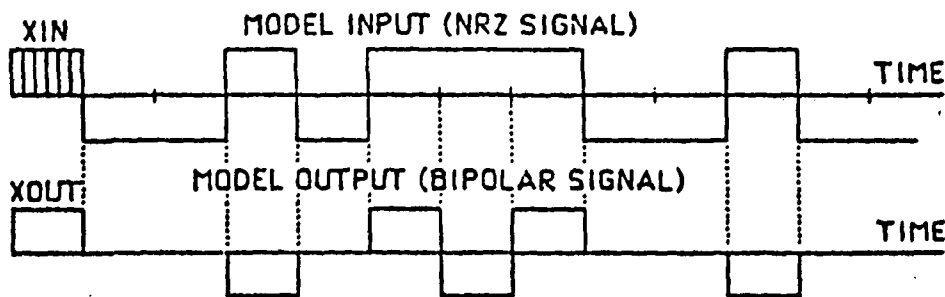
F-4 These new software features heighten BOSS' utility, compared with that of earlier simulators. For instance, to create a module for converting NRZ waveforms to bipolar (Figure 4a), SYSTID requires the user to write a FORTRAN-like subroutine model describing the module (Figure 4b). BOSS, on the other hand, lets the user construct the model as a block diagram (Figure 4c), with the Code Generator producing the FORTRAN.

Not only is BOSS' diagrammatic representation closer to the hardware realization than the code description, it also "represents more clearly the module's conversion function," Manning pointed out. BOSS also eliminates the possibility of user-introduced code errors. And, finally, the BOSS diagram is faster to create.

As the operating system for SCSS, BOSS faces a tough challenge in modeling the Space Station's communication links. The Station will serve as a central node in a major multiple access communications network serving numerous, diverse spaceborne users, ranging from the space shuttle to unmanned spacecraft and "spacewalking" astronauts in extravehicular activity (EVA), working near the Station. In addition to being widely distributed in several regions about the Station, these users will employ a variety of data rates and waveforms.

F-5 Figure 5 illustrates several key features of this multiple-access

ORIGINAL PAGE IS
OF POOR QUALITY



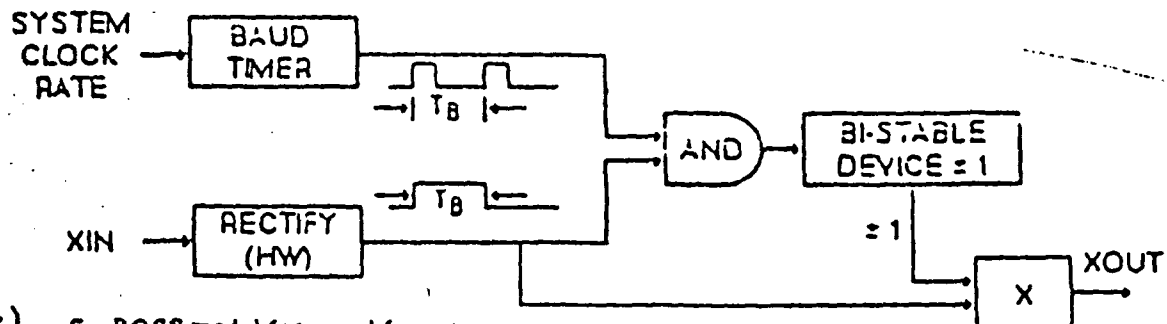
- a) A. NRZ to Bipolar Signal Format Conversion. In Bipolar format, binary zeros are represented by zero amplitude pulses and binary ones by pulses with alternating polarities.

```

MODEL. XIN < BIPOLAR (NSAMPL) > XOUT
COMMON: TIME,DT
STACK<1> I FIRST, I LAST
INITIALIZE:
  IF (I FIRST.NE.0) GO TO 10
  I LAST = -1
  I FIRST = 1
10 CONTINUE
SIMULATE:
  IF (XIN.LT.0.0) XIN = 0.0
  ZTIME = TIME/DT
  NCOUNT = MOD (ZTIME,NSAMPL)
  IF ((NCOUNT.EQ.1).AND.(XIN.GT.0.0))
    I LAST = I LAST * (-1.0)
  XOUT = XIN * FLOAT(I LAST)
END:

```

- b) B. SYSTID-like program for signal format conversion (Source: Reference [1])



- c) C. BOSS model for signal format conversion

Fig. 4 Simulation Models for a Signal Format Converter

architecture, as reflected in the Station hardware configuration. These include the dispersed placement of antennas, necessitated by the large, extended Space Station structure and the requirement to provide coverage in multiple surrounding zones. Each antenna is part of the receiver/transmitter equipment that provides low noise amplification/power amplification, down- and upconversion, filtering, and switching. Typically, the receiver/transmitters are connected to signal processors by means of the signal distribution system, which consist of data buses of coaxial or fiber optic cables. Data buses have several architectural options, including one cable per antenna cluster and one cable per signal, as well as baseband buses and intermediate frequency frequency-division-multiplexed buses. The bus-linked signal processors provide modulation/demodulation, bit synchronization/detection, and coding/decoding functions.

At present, a frequency division multiple access (FDMA) scheme is the leading candidate for the Stations communication architecture, according to Peter Nilsen, a communication systems consultant who has worked on TRW's Space Station program. Links will carry audio, video, command, and telemetry data.

SCSS will play an important role in evaluating performance and design trade-offs for the Station's MA links, particularly return links, which are subject to the challenge of receiving simultaneous transmissions from near and far users. "For example, simultaneous transmissions from an Orbital Maneuvering Vehicle 37 kilometers from the Station and an EVA astronaut working only 2 meters from a receive antenna would result in 85 dB of

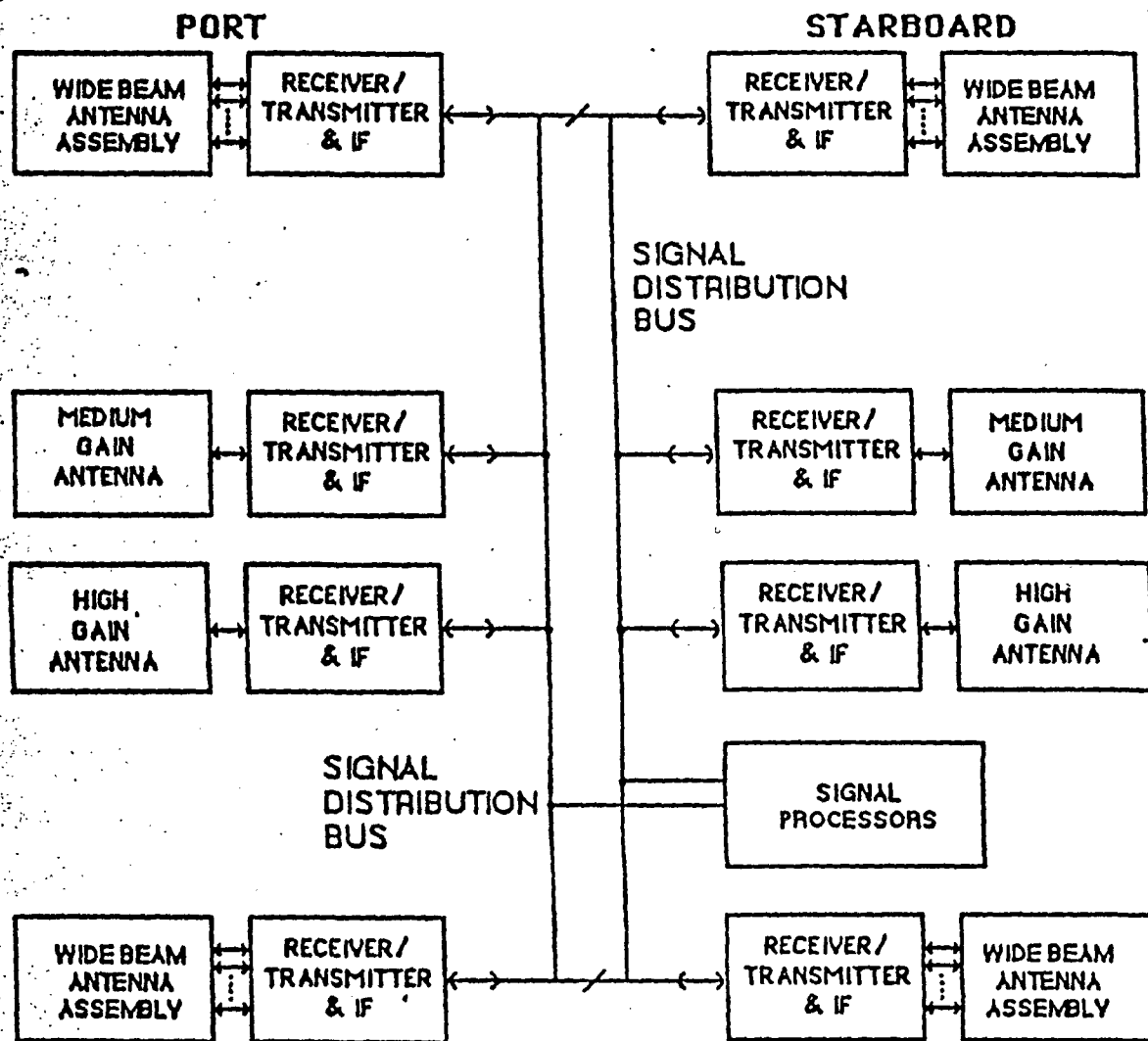
variation in signal strength at the receiver, due to range alone," Nilsen said. Such situations can result in potential adjacent channel interference, as well as the creation of intermodulation products in the receiver. SCSS simulations will help to evaluate the types of degradation resulting from near/far transmissions; they will also help in evaluating filter designs and channel spacings used to counter the near/far problem, Nilsen added.

To ensure that BOSS simulation will be available for these sorts of tasks, the SEL has formulated a two-phase approach to development of SCSS, Manning said. A 6-month Architecture Definition phase will identify Space Station simulation requirements, determining such items as the communication links to model, the devices contained in each link, and the communication schemes involved. While aspects of this communication system will certainly change as space station design proceeds, Manning noted that TRW's development of SCSS is structured to accommodate new requirements and features as they arise.

In the succeeding Development phase, the SEL will implement SCSS as a functioning simulator, with Space Station models determined during the Definition phase included as part of the SCSS library. Top-level models of Station transmitters and receivers and user transceivers will be constructed by connecting appropriate models on the graphics display. For example, a user transmitter model connected to a channel model and then to a Station receiver model would form a typical link (Figure 7). Links can then be verified by inspecting time and frequency response, estimating BER using semi-analytical and Monte Carlo simulations, and by estimating

TRW

UPPER BOOM



LOWER BOOM

Figure 5. Space Station Generic Multiple Access Communication System

acquisition performance. BOSS allows rapid reconfiguring of any link in response to changes in the communications architecture.

SCSS' hardware, as well as its software modules, contributes to its adaptability. The computer hosting SCSS will be a Digital Equipment Corporation VAXstation-IITM, which incorporates sophisticated graphics, multi-windowing capability, and a processing speed 90 percent that of a VAX-11-780. The VAXstation-II's compatability with all VAX-11/750/780 computers enables it to network with Johnson Space Center computers and to utilize data from NASA's Dynamics Simulation System (DSS), which simulates the effects of movement or positional relationship on communications links between the Space Station and other spacecraft in communication with it. The DSS link will supply the SCSS with parameters such as range loss, doppler shift, and multipath loss, for use in dynamic simulations.

The local area network interconnecting SCSS with DSS will consist of ETHERNET hardware, controlled by DECNET software, and a DELNI interconnect device and cabling (Figure ⁶8). "The fact that SCSS runs on its own computer is a tremendous advantage to NASA, since this allows DSS and SCSS to run concurrently. A conventional simulation running on a NASA VAX can't match this boost in throughput," Manning said.

The DSS interface will permit SCSS to operate in three modes. Mode 1 is the power-normalized operation common in most simulation packages. The user sets device operating points, such as amplifier input drive levels. Specific power levels at any device are irrelevant; only the level relative to interference and noise source levels is important. This mode is useful

TM

Registered Trademark of Digital Equipment Corporation

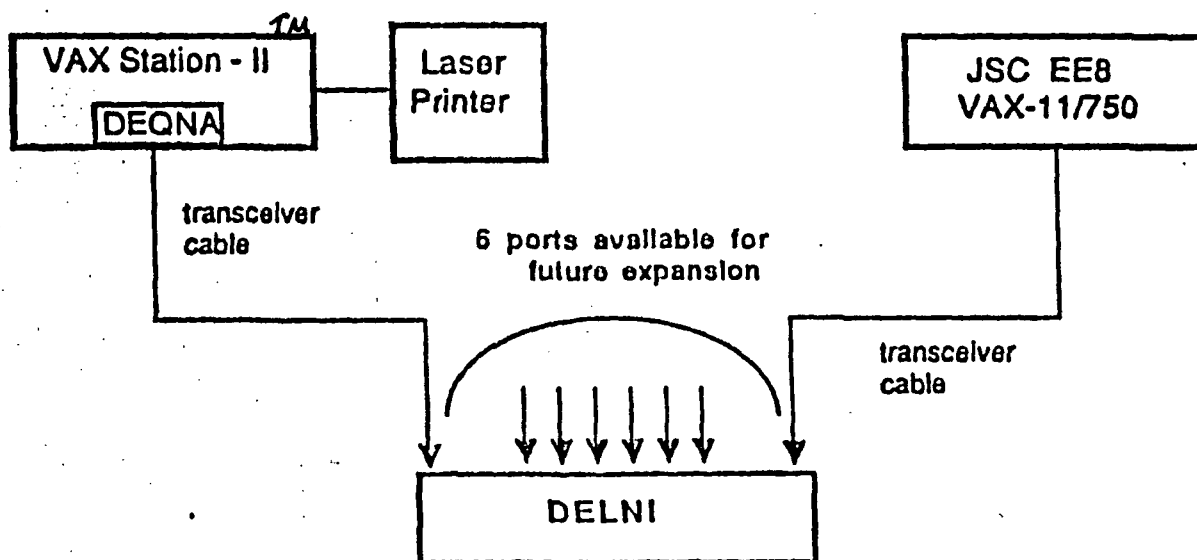


Figure 6 *LAN linking SCSS with DSS*
 TRW-Proposed Local-Area-Network-Requires-No-DSS-Modifications

TM Registered Trademark of Digital Equipment Corporation

for determining signal distortion measures (from filtering and nonlinear amplification) and for performing power-normalized analyses such as acquisition or BER sensitivity analysis.

Mode 2 operations more closely approximate a real system in that constant gains or attenuations are explicitly included in the channel definition. Input level as well as device characteristics determine device output levels. This mode is useful for determining circuit margins and performing other level-specific analyses.

SCSS supports Mode 2 by supplying models that alter power levels (gains or attenuations); including level parameters in models such as filter insertion losses; and including models which have their operating points set by their input level (e.g., nonlinear amplifiers). The user simply includes these models in his simulation definition.

Mode 3 is functionally identical to Mode 2 with the added requirement that data for certain models must be imported from the DSS. DSS outputs are time-sampled values of the following quantities: doppler frequency shift, antenna gains, and polarization, pointing, range, and multipath losses. Like DSS, SCSS is a time-domain simulation, that is, at discrete points in time, each simulation parameter is calculated and the clock updated. This procedure matches the output format of DSS's data except for the detail of sampling interval compatibility. SCSS is designed to convert DSS outputs to a compatible sampling interval.

To include DSS output data in a simulation, SCSS possesses models in its

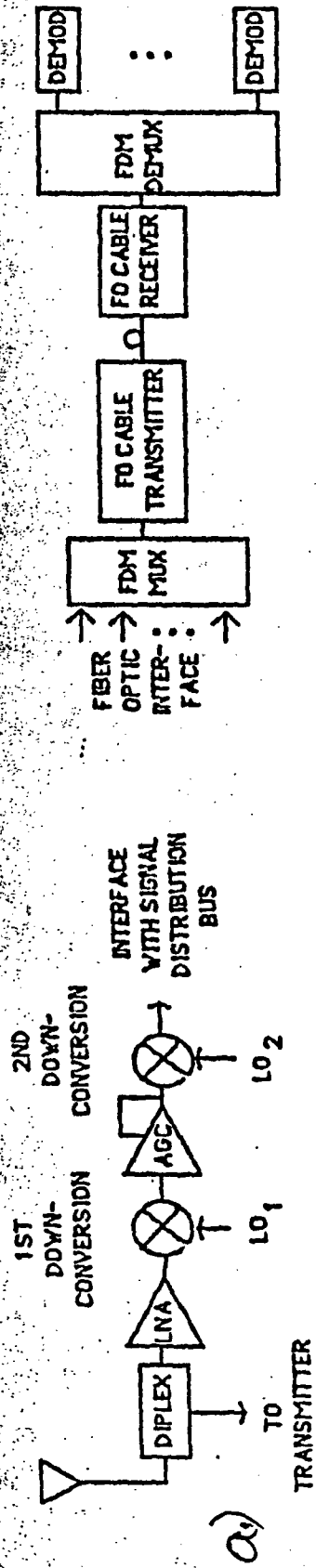
library for each type of DSS output. These models access the DSS data files from a NASA VAX 11/750 and incorporate the data into the communications simulation definition. Then the simulation proceeds as in Mode 2.

SCSS will employ these three modes in simulations to evaluate numerous Space Station design options. For instance, SCSS may help to select between two options for configuring the FDMA receivers, as shown in Figure 1a. Option A proposes block downconversion, i.e. all return link carriers received by a given antenna are downconverted together in the same downconverter, amplified in the same amplifier, and sent to the demodulators via a common cable. This option minimizes the hardware required. Receiver performance, however, both BER and acquisition, may suffer from severe intermodulation products generated in the downconverters, amplifiers, and fiber optic transducers (since fiber optics is a configuration option).

As part of a definition study for the Station's communications and tracking system, ESG's Space Station program used LINK to simulate a configuration similar to Option A, Nilsen said. That study indicated that usually only carriers immediately adjacent to the desired carrier produce undesirable effects in the simulated receiver. However, the situation differs when a nonadjacent carrier signal is substantially stronger than the desired carrier, a likely occurrence with the Space Station.

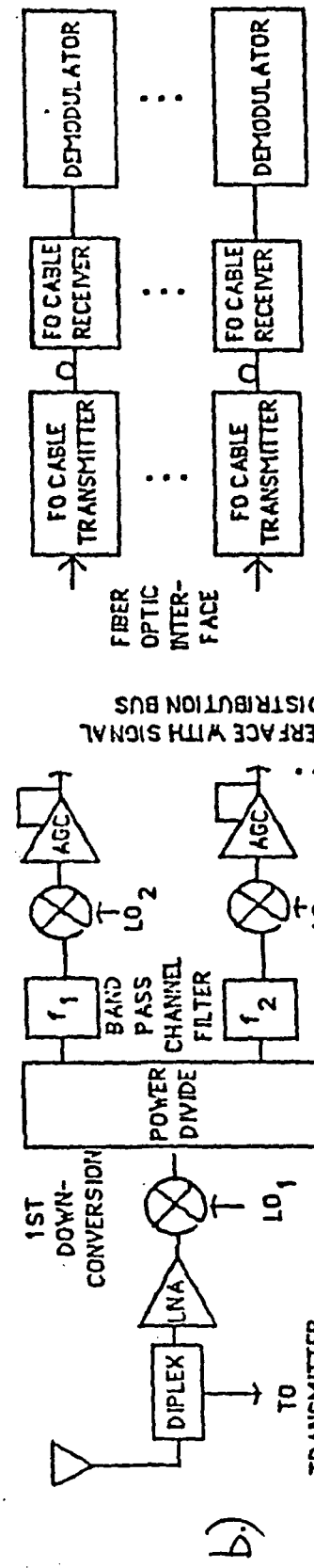
To improve computational efficiency for calculating multiple carrier situations, SCSS simulation will characterize the output of each block shown in Figure 1a whenever the situation occurs. This characterization

ORIGINAL PAGE IS
OF POOR QUALITY



OPTION C: MULTIPLE RETURN LINK SIGNALS
PER FIBER OPTIC (FO) CABLE

OPTION A: BLOCK DOWNCONVERSION
CHANNELIZATION AT DEMODULATORS



OPTION D: ONE RETURN LINK SIGNAL
PER FIBER OPTIC (FO) CABLE

OPTION B: CHANNELIZED DOWNCONVERSION

FIGURE 7 SCSS FLEXIBILITY ALLOWS OPTIONS FOR LINK MODELS

will then become part of the database of each of the block models.

Subsequent simulations involving these blocks will not have to reconstruct the multiple-carrier effects.

Fig 1 The alternative design, Option B (Figure 1b), channelizes each return link carrier into its own intermediate frequency channel via channelizing filters. This approach greatly reduces the intermodulation problem associated with block downconversion and allows automatic gain control on each individual carrier -- but, at a penalty in extra hardware. Simulating option B with SCSS is merely a matter of rearranging the icons and changing the parameter values for the models in option A on the graphics terminal, since both options use most of the same elements.

NASA will need to determine the costs versus the benefits involved in both implementations, Nilsen said. "SCSS will be a powerful and valuable tool in evaluating the power spectral densities and bit error rates obtained with each of the two approaches."

BOSS' developers hope that the simulator's visibility as the operating system of NASA's SCSS will highlight its modular versatility. For one thing, BOSS is highly cost-effective when compared to the labor needed to develop a special purpose simulator, Manning said. Development of a simulator for acoustic wave demodulators on a TRW space communications program (carried out before BOSS became available) required about 12 person-months. Four main tasks -- developing the basic simulator capability, creating a user-friendly input processor, debugging and verifying code, and adding new device models and simulator features --

required about 3 months each.

Use of BOSS for the same simulation would have required only a sixth of that time, based on current estimates. "BOSS completely eliminates the tasks of developing a user friendly input and debugging and verifying code," Manning said. "It also greatly reduces the time spent learning the basic simulator capability. And adding new devices involves only configuring a few modules to perform the desired function."

While the immediate focus of attention is on development of SCSS, BOSS' general-purpose architecture promises numerous other applications for TRW and its customers, Rey noted. Since BOSS simulates at the functional level, it furnishes hardware engineers an easy means to verify system configurations. Therefore, the SEL is working to introduce BOSS as part of ESG's Computer Aided Engineering Process, a Group effort aimed at furthering automation of hardware design and development. "We also plan to apply BOSS communications system simulation to all SCD and DCD space communications programs," Rey said.

The SEL is currently buying VAXstations to equip a systems simulation facility scheduled to begin operations in 1987, Rey added. A Microvax computer linked to the VAXstations will store a large database containing models for SCD and DCD communications systems. In addition, the facility will have links to a hardware database established in SCD's Digital Systems Design Center. "Access to the Center's database means we can put real hardware data into our systems simulations," Rey said. "That sort of

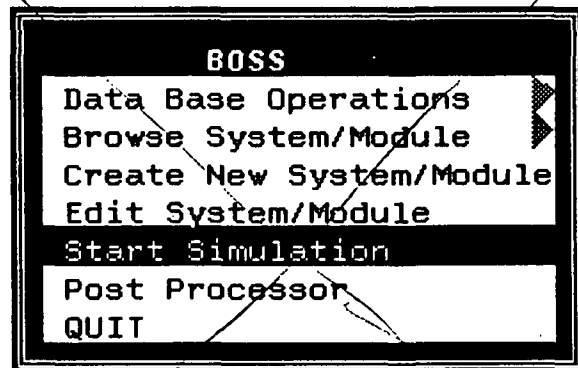
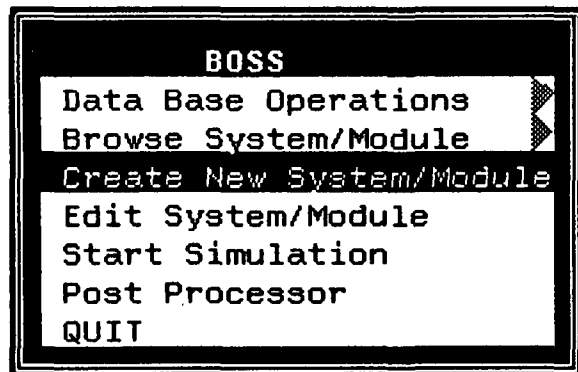
database means we'll be able [using BOSS] to perform rapid prototyping of communications systems."

The systems simulation facility will also enable TRW to efficiently perform quick simulation studies for systems engineering customers. "This capability will better position us to support ESG customers in advanced system design," Rey added.

BOSS' general purpose capabilities interest engineers throughout TRW's Electronics and Defense Sector, who see possibilities for non-communications applications. "We've started working with people from the Space and Technology Group on simulating control systems and power systems. BOSS is ideal for analyzing the stability of spacecraft power subsystems, with their long cable runs and feedback loops," Rey said. He added that BOSS has also brought inquiries from the Defense Systems Group.

The widespread interest in BOSS' potential is no surprise, said Wiswell. "Hardware has become too complex and expensive for us to build breadboards everytime we design a new system. Simulation is the way to go, and there's going to be more and more of it."

--END--



ORIGINAL PAGE IS
OF POOR QUALITY

Creating and Simulating
a QPSK Transmitter:
The Main Steps as They
appear on ^{the} BOSS screen.

QPSK TRANSMITTER

ORIGINAL PAGE IS
OF POOR QUALITY

could be
deleted

QPSK TRANSMITTER

Edit Menu

- Add Module
- Add Signal Input
- Add Signal Output
- Modify Parameter Description
- Save Incomplete
- Save New System/Module
- Quit

ORIGINAL PAGE IS
OF POOR QUALITY

QPSK TRANSMITTER

☐ RANDOM DATA ▶

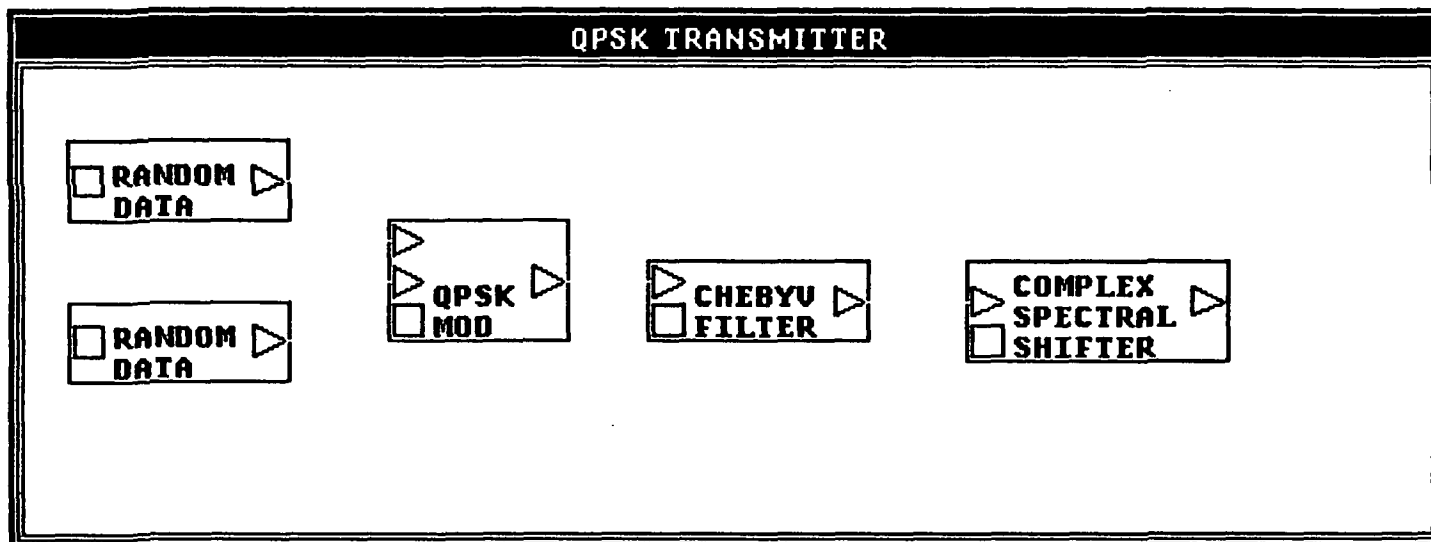
Module Menu

- Copy
- Delete
- Expand
- Move
- Options

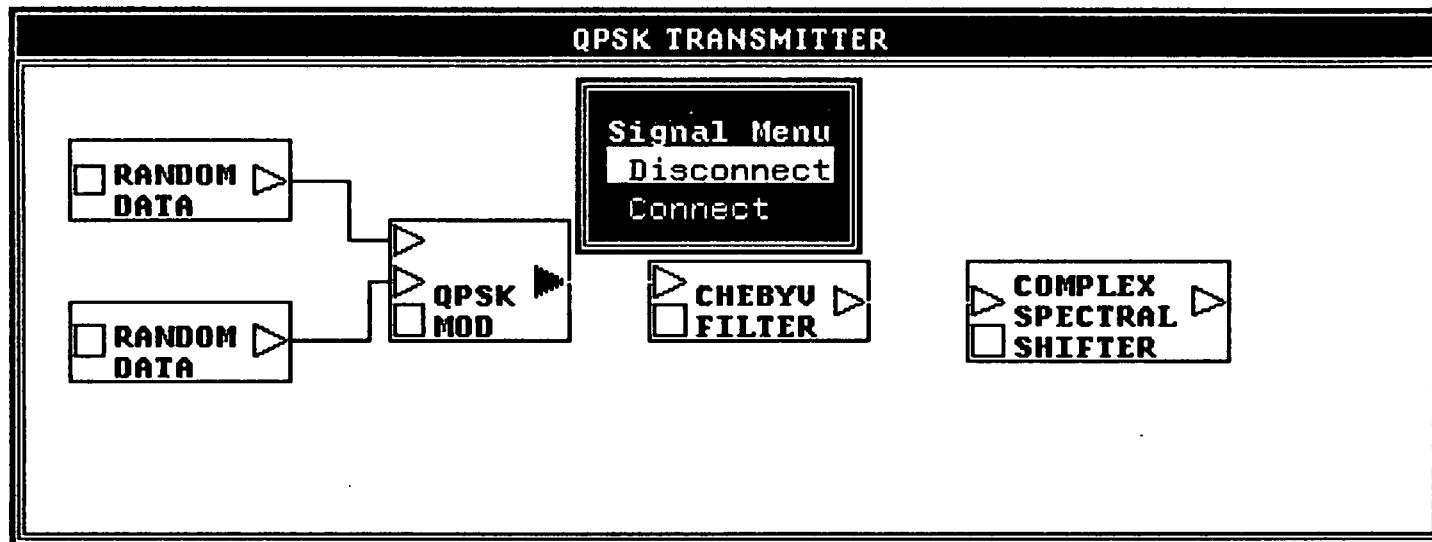
Other Options Menu

- Replicate
- Label
- Add Enable Signal

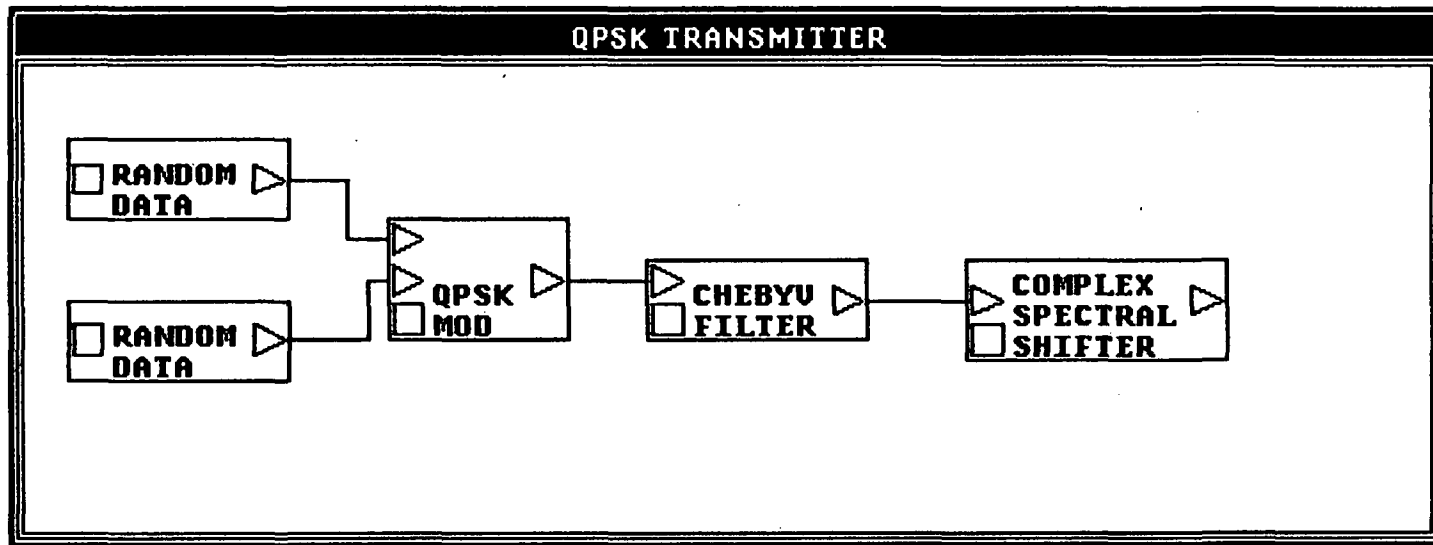
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY



Edit Menu

Add Module

Add Signal Input

Add Signal Output

Modify Parameter Description

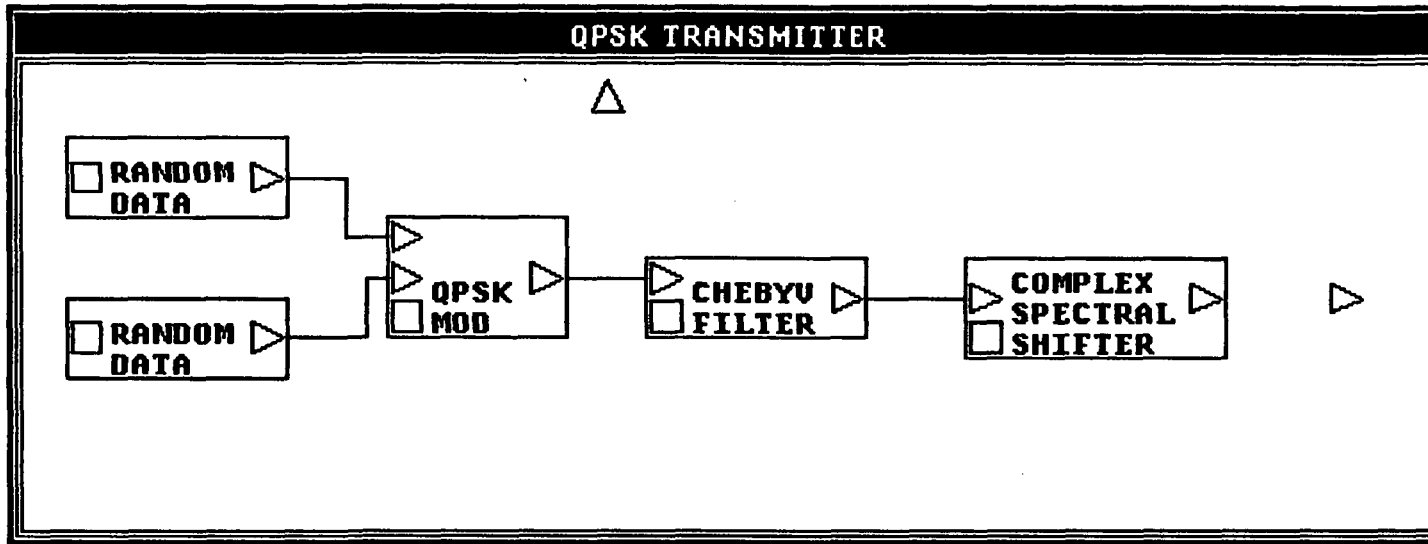
Save Incomplete

Save New System/Module

Quit

ORIGINAL PAGE IS
OF POOR QUALITY

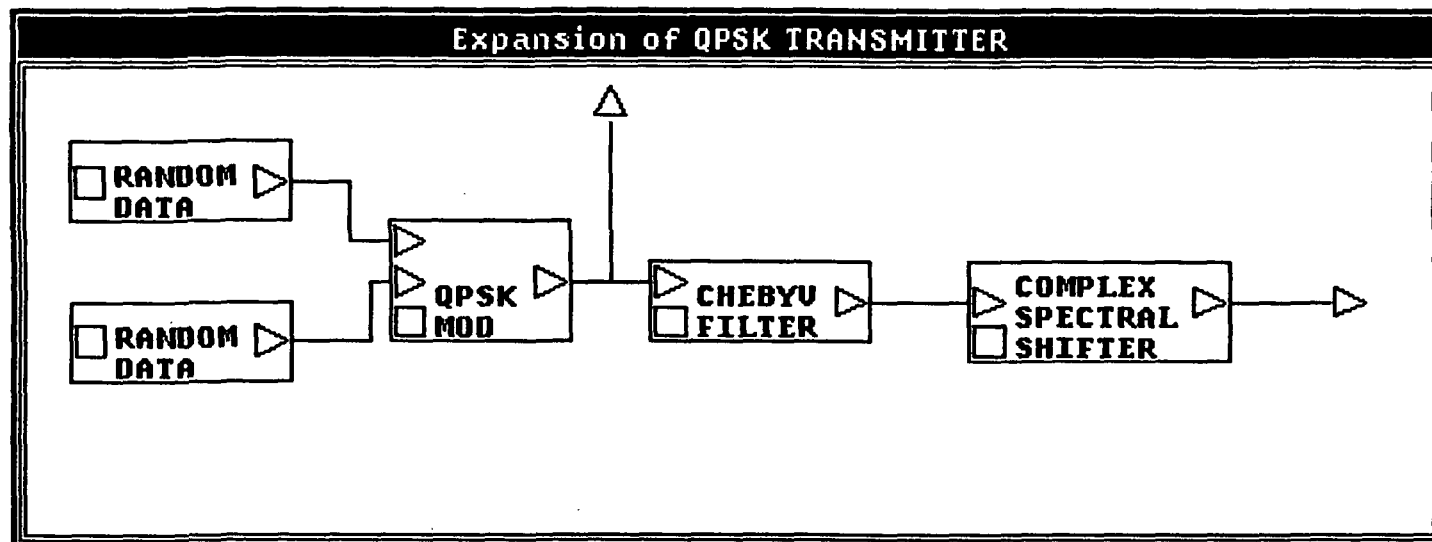
*Could be
deleted*

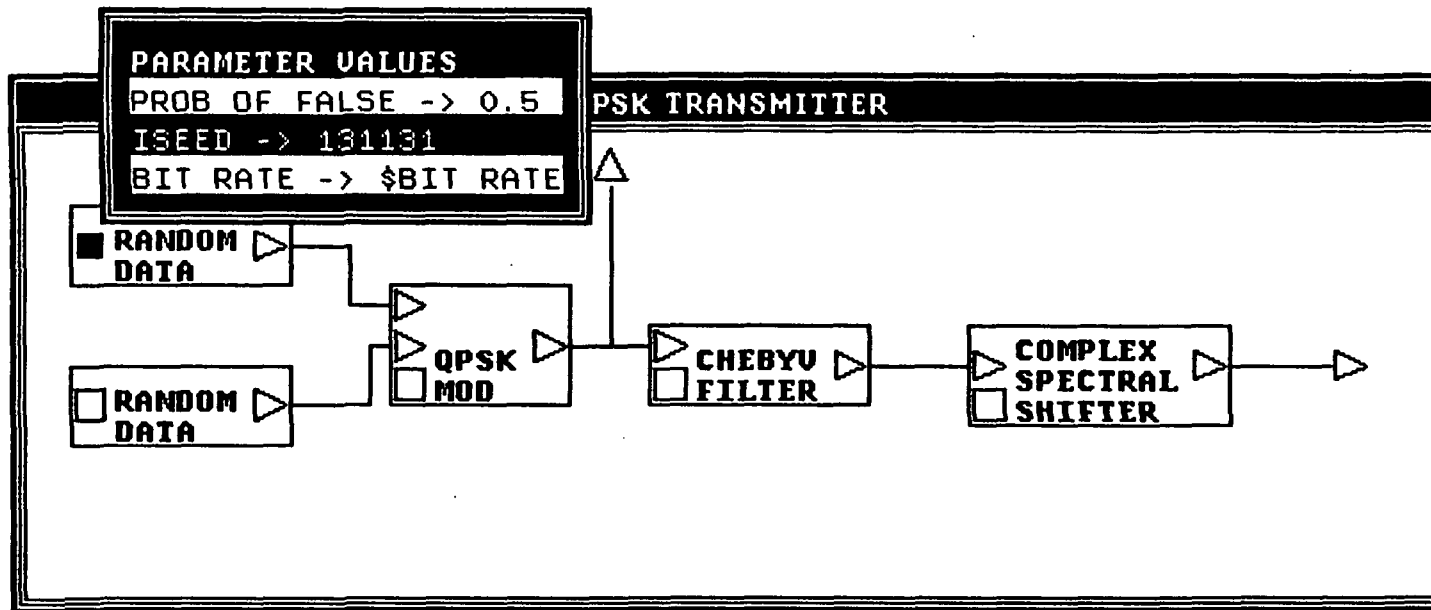


ORIGINAL PAGE IS
OF POOR QUALITY

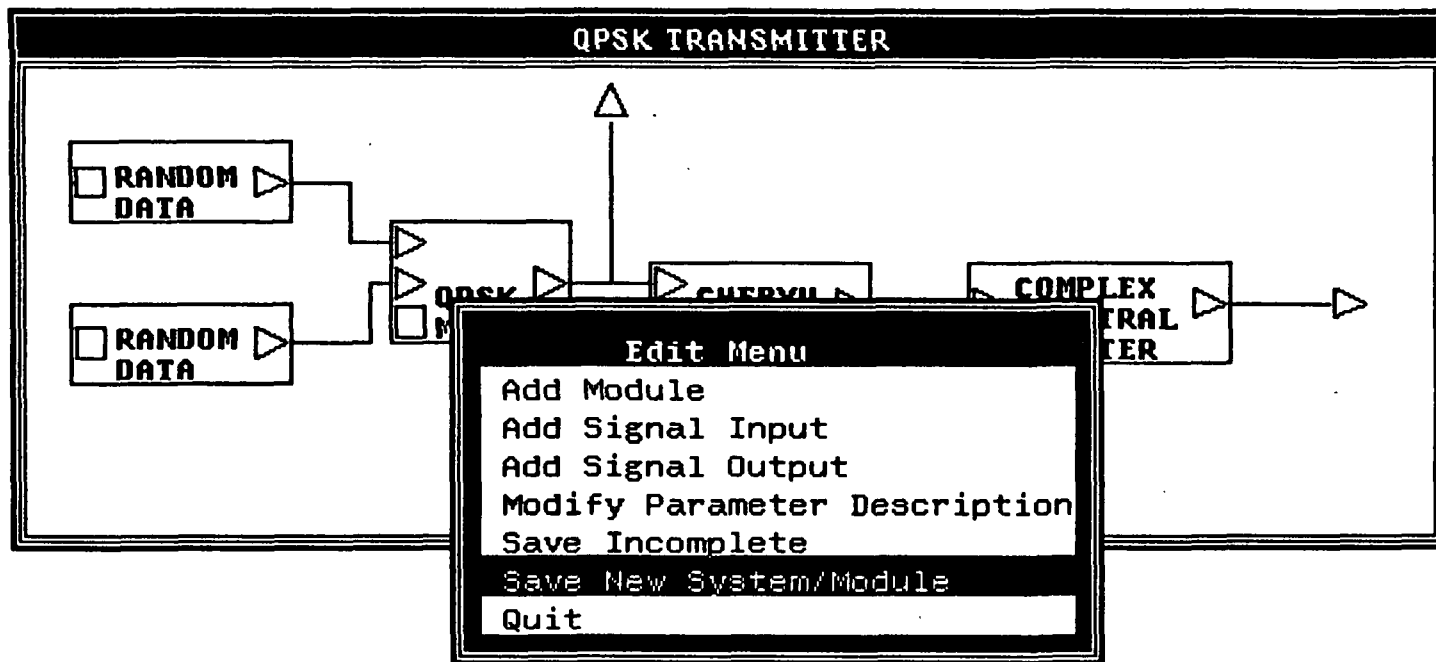
could be deleted
if "Signal Out put" added
option added
to row #10

(9)





ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

BOSS

Data Base Operations

Browse System/Module

Create New System/Module

Edit System/Module

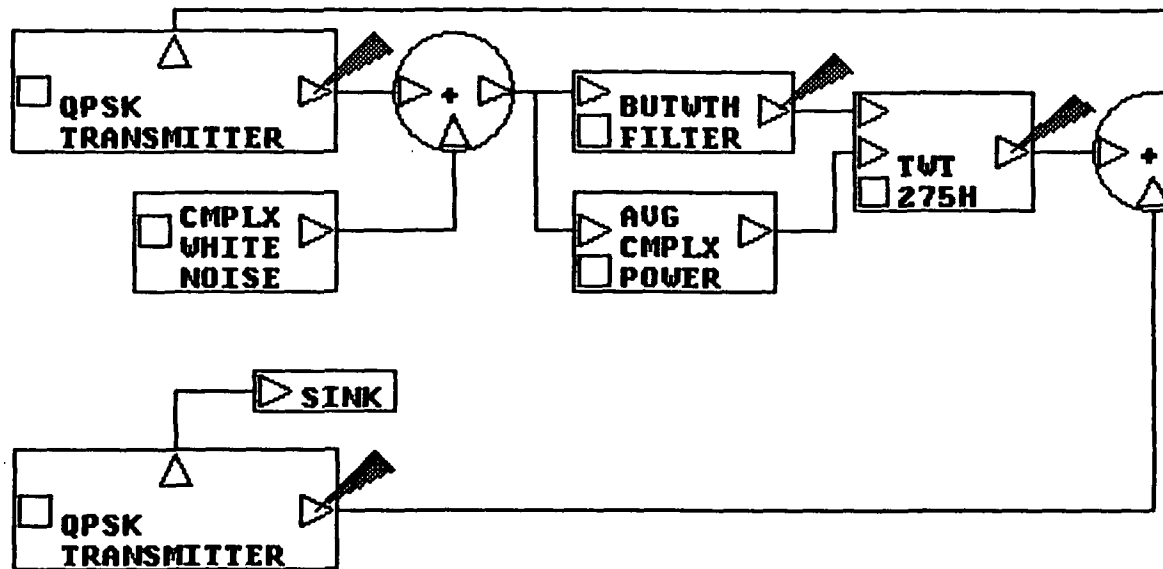
Start Simulation

Post Processor

QUIT

ORIGINAL PAGE IS
OF POOR QUALITY.

ANALYTIC QPSK W/INTERFERER -

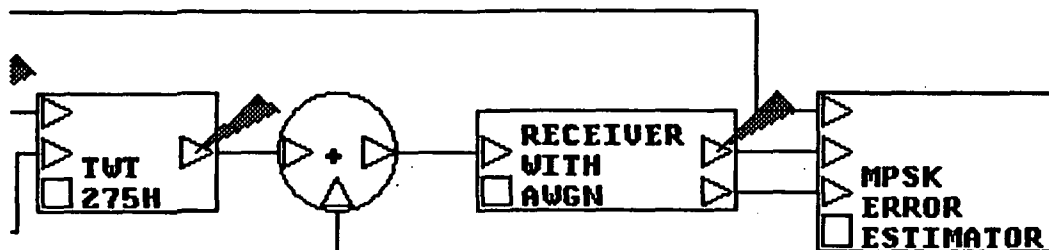


ORIGINAL PAGE IS
OF POOR QUALITY

2 alternatives
presented

142

SK W/INTERFERER - RUN1



PARAMETER VALUES

STOP-TIME -> 1000.0
DT -> 0.1
TWT BACKOFF -> -3.0
CHANNEL FILTER BANDWIDTH -> 1.0
UPLINK NOISE POWER -> 0.2
SAMPLES/SYMBOL -> 10
SYMBOLS FOR ERROR ESTIMATION -> 300
INTERFERER BIT RATE -> 1.0
BIT RATE -> 1.0
TRANSMIT FILTER BANDWIDTH -> 1.0

ORIGINAL PAGE IS
OF POOR QUALITY

101 b

Simulate Menu
Simulate
Set Parameters
Quit

ORIGINAL PAGE IS
OF POOR QUALITY

Paste up on
Screen that shows
parameter values

(14c)

BOSS

Data Base Operations

Browse System/Module

Create New System/Module

Edit System/Module

Start Simulation

Post Processor

QUIT

ORIGINAL PAGE IS
OF POOR QUALITY

PROCESSING COMMANDS

Time_Plot
Eye_Diagram
Scatter_Plot
Correlation
Covariance

Spectrum

Differential_Spectrum

General_Plot

Histogram

Tabular_Plot

View Printed Output

System_Block

Create_Viewgraph

Open System

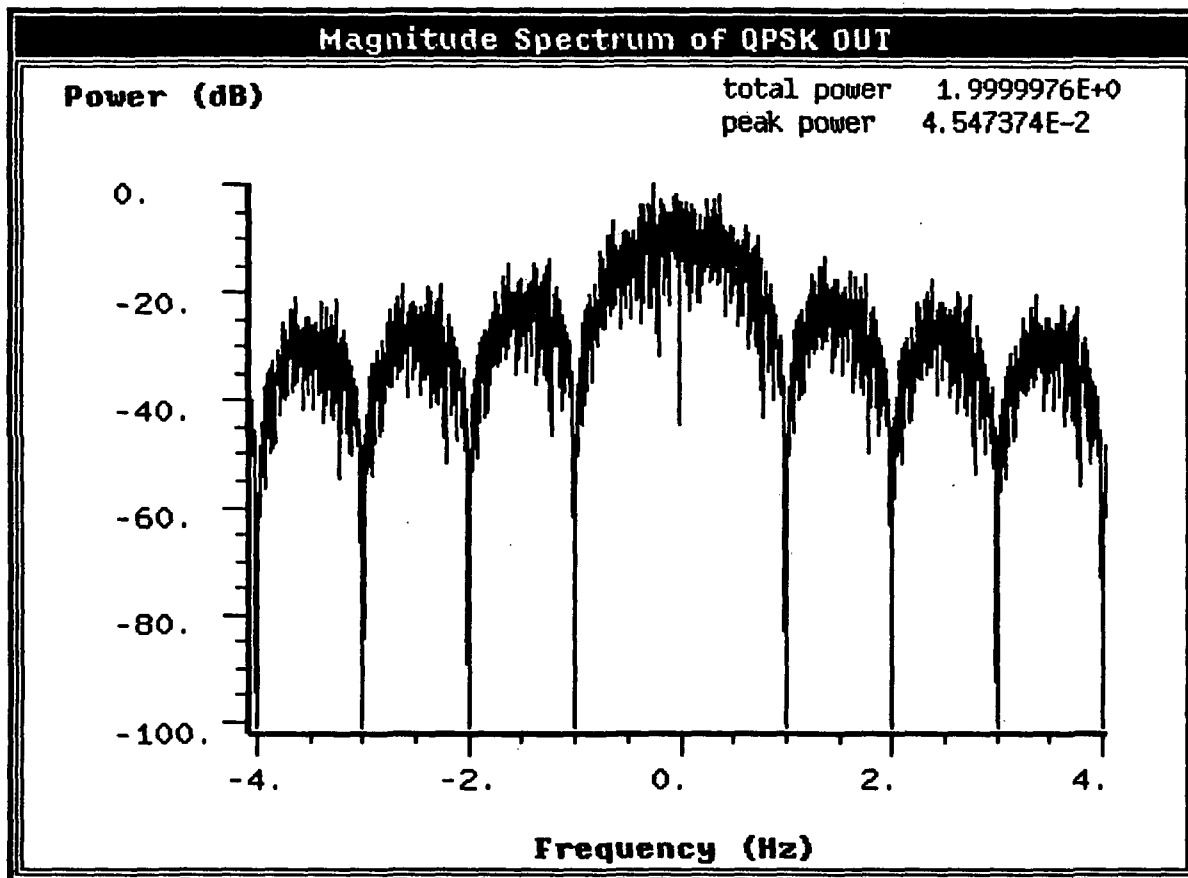
Set Plot Sizes

QUIT

Magnitude_Spectrum

Phase_Spectrum

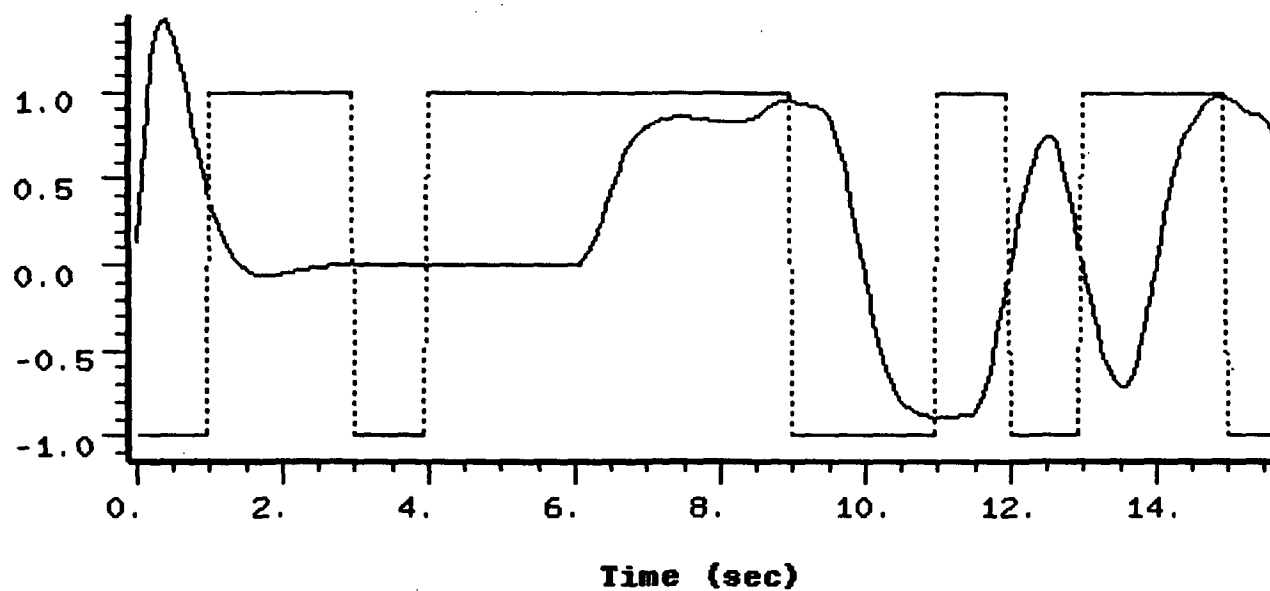
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

Time Plot of REC FILT OUT

real part



Sidebar

TRW's University Connection

TRW's collaboration with the University of Kansas (KU) in developing BOSS highlights the success of ESG's 5-year-old University IR&D Program. Through the program, ESG funds university researchers to augment the Group's capabilities in basic research. "It enables us to get some good research done in areas where we have ongoing IR&D work. We also seek tighter ties between individual faculty members, working in technical areas of interest to us, and their counterparts at TRW," said ESG Chief Scientist Dick Booton.

To assist in developing BOSS, Eric Wiswell, head of the TRW IR&D effort, called on KU's Professor Sam Shanmugan, who he called "probably the United States' preeminent authority on communications systems simulation." In 1986, the Institute of Electrical and Electronic Engineers honored Shanmugan for his contributions to computer-aided modeling and analysis by electing him as a fellow.

Sciences Shanmugan heads the university's nationally renowned Telecommunications *and Information* Laboratory, which carried out the IR&D work. The collaboration called for TRW to direct the overall effort and provide IR&D funding. TRW also developed the hardware models that comprise the simulator libraries. KU provided an initial architecture study report and recommendations on hardware programming languages for the new simulator. After development began, KU performed much of the software implementation, participated in design reviews, and ultimately delivered a working version of BOSS to TRW.

During system development, the Company provided the lab with a DEC Vaxstation and later donated the University a Vaxstation-II. TM

With the program underway, "we found that TRW and the KU team saw pretty much eye to eye on the main issues of simulator development," Wiswell said. He added that KU had a knowledge and research base in artificial intelligence and systems simulation that TRW does not possess. For his part, Shanmugan said that "BOSS gave the lab an opportunity to work on a challenging, 'real world' problem. And it offered a chance for our students to learn through interaction with working engineers."

In 1986, KU was one of 12 universities across the USA receiving university IR&D funds for work on 20 separate projects. Total funding totalled \$600,000, with projects receiving from \$10,000 to \$50,000 each, Booton said.

In addition, an ESG-funded Fellowship Program at KU contributes to the Group's strength in communication systems simulation. The fellowships support graduating seniors who go on to pursue master's degrees in science or engineering at the university. Fellowship recipients work their senior summer at TRW's Space Park, and are offered full-time employment upon finishing their master's. The program has brought more than 20 new employees to ESG, including Jerry Lacy, Dodge Johnson, Darin Haeger, Ed Friedman, and Paul Snow, all members of the SEL's technical staff.

TM Registered Trademark of Digital Equipment Corporation